



Conceito

Oque é um Banco de Dados?

Um Banco de Dados ou SGBD (**Sistema Gerenciador de Banco de Dados**) é uma ferramenta que armazena e administra as informações utilizadas por diversas aplicações de software. Todo sistema, por mais simples que seja, precisa acessar um Banco de Dados.

As informações que ficam guardadas no Banco de Dados podem ser dos mais variados tipos: nomes e endereços de clientes, lista de serviços prestados, volumes de vendas, resultados financeiros. Além de uma série de outros elementos que fazem parte da estrutura de empresas de todos os portes e setores da economia.

Esse modelo de armazenamento não é exatamente uma novidade. Os dois grandes nomes desse mercado: o **SQL Server da Microsoft** e o **Oracle Database** já têm décadas de existência. O primeiro SGBD a ser utilizado no mundo, o Oracle, surgiu no final dos **anos 70**. Já o SQL Server foi lançado pela **Microsoft em 1988**.

Desde então, eles vêm sendo constantemente atualizados e adaptados às novas tecnologias. A cada atualização, os sistemas ampliam suas funcionalidades e aumentam as possibilidades de uso por um número ainda maior de empresas em todo o mundo.

Conceito

Há diversas opções de Banco de Dados, direcionadas para públicos distintos e para diferentes demandas de volume de trabalho. Existem variações em relação a parte técnica (**linguagens e compatibilidades com determinadas plataformas**) e também funcional (**modelos relacional, plano ou hierárquico**). Eles podem ser, por exemplo, para trabalhos locais ou em nuvem, e oferecem edições de acordo com o porte e as necessidades específicas das empresas.

Um empreendedor de pequeno porte, com um negócio que gera poucos arquivos e necessita de um nível de processamento mais limitado, pode optar por uma edição mais básica de um Banco de Dados, e assim não precisa fazer um investimento tão alto.

Já uma empresa maior, que precisa de serviços Big Data, ou seja, com imenso volume de dados, tem a necessidade de mais velocidade para processamento de mais informações e versões mais robustas do Banco de Dados.

Servidor de Banco de Dados

Oque veremos no decorrer?

- **Instalar Ubuntu Server do zero**
- **Configurar endereço IP, nome de maquina e domínio**
- **Habilitar mysql e ssh no Firewall**
- **Atualizações do sistema**
- **Configurar acesso remoto ao MySQL**

Início básico

De início, vamos baixar a imagem a ser instalada, localizada no endereço a baixo:

<https://releases.ubuntu.com/24.04/ubuntu-24.04-live-server-amd64.iso>

A iso tem aproximadamente **2.5GB** e vai demorar um pouco. Assim que o download estiver concluído, vamos a instalação na VM.

Recomendo que em memória, seja disponibilizado mínimo **de 4GB** e em disco mínimo **de 40GB**.

- Assim que a VM subir com a instalação do sistema, vamos dar início as configuração de rede, onde selecionamos configurar e vamos adicionar um IP fixo(**Servidores sempre IP fixo por questões óbvias**). O nome da rede sera **SGBD.com.br**

Início básico

Assim que o sistema estiver terminado a instalação, vamos reiniciar conforme solicitado e dar início a instalação do bando de dados.

Atualizando o sistema:

```
sudo apt update && sudo apt upgrade -y
```

Próximo passo é criar a exceção no Firewall:

```
sudo ufw allow mysql && sudo ufw allow ssh
```

Início básico

Feito isso vamos a instalação do MySQL:

```
sudo apt install -y mysql-server
```

Habilitando o serviço:

```
sudo systemctl enable --now mysqld
```

Vamos criar uma senha segura e remover possíveis fragilidades do MySQL:

```
sudo mysql_secure_installation
```

Início básico

Acessando o Banco de Dados:

```
mysql -u root -p
```

Verificando a base de dados:

```
show databases;
```

Tudo funcionando.

Agora vamos garantir acesso remoto ao nosso banco de dados.

Primeiro precisamos editar o arquivo de configuração do mysql para que tenhamos como acessá-lo de forma remota:

```
sudo vim /etc/mysql/mysql.conf.d/mysqld.cnf
```


Início básico

Localize a linha:

Bind-address = 127.0.0.1

Altere para:

Bind-address = 0.0.0.0

Salva a alteração e reinicie o mysql:

sudo systemctl restart mysql

Início básico

Agora, vem a parte de criação de usuário para o acesso remoto do db, e para isso precisamos acessar o mysql:

```
sudo mysql -u root -p
```

Criar o usuario para o acesso:

```
create user 'seu_usuario'@'%' identified by 'sua_senha';
```

Conceda ao usuário os direitos para criar novos usuários e a capacidade de recarregar os direitos globais com:

```
grant create user, reload on *.* to 'seu_usuario'%';
```

Início básico

Conceda todos os direitos adicionais ao novo usuário, juntamente com o direito de conceder novos direitos a outros usuários:

```
grant all on *.* to 'seu_usuario'@'%' with grant option;
```

Conceda privilégios de criação de banco de dados e seleção:

```
grant create, select on *.* to 'dbadmin'@'%' ;
```

Atualizar os privilegios:

```
FLUSH PRIVILEGES;
```

```
exit
```

Início básico

Acessando o banco com o usuário criado:

```
mysql -h 10.10.10.25 -u seu_usuario -p
```

```
show databases;
```

```
use db_turminha;
```

```
show_tables;
```

Exercício:

Criar múltiplas tabelas que se relacionem entre si:

Exercício

```
CREATE TABLE Usuarios (  
    Id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,  
    Usuario VARCHAR(32) UNIQUE,  
    Senha VARCHAR(32),  
    Nome VARCHAR(100),  
    Estado CHAR(2),  
    Cidade VARCHAR(100),  
    Email VARCHAR(255),  
    Nascimento DATE,  
    Ultimo_Acesso DATETIME,  
    Saldo DECIMAL(10,2),  
    Biografia TEXT,  
    Status ENUM('Ativo', 'Inativo')  
);
```

Exercício

Detalhe dos tipos de dados para cada campo e explicar as razões para as escolhas:

Id: `INT`, ideal para identificação única do usuário.

Usuário: `VARCHAR(32)`, para armazenar nomes de usuário, permitindo até 32 caracteres.

Senha: `VARCHAR(32)`, para armazenar senhas, também permitindo até 32 caracteres.

Nome: `VARCHAR(100)`, flexível para acomodar nomes de diferentes comprimentos.

Estado: `CHAR(2)`, adequado para siglas de estados de tamanho fixo.

Cidade: `VARCHAR(100)`, para acomodar nomes de cidades de diferentes tamanhos.

Email: `VARCHAR(255)`, um padrão comum para endereços de email.

Exercício

Nascimento: `DATE`, eficiente para armazenar apenas datas.

Ultimo_Acesso: `DATETIME`, ideal para registrar informações temporais precisas.

Saldo: `DECIMAL(10,2)`, preciso para valores monetários.

Biografia: `TEXT`, apropriado para armazenar grandes quantidades de texto.

Status: `ENUM('Ativo', 'Inativo')`, eficiente para campos com um conjunto limitado de valores.

Id: `INT UNSIGNED AUTO_INCREMENT PRIMARY KEY`. Um número inteiro positivo que se autoincrementa e é chave primária, ideal para identificação única de cada usuário. Este campo pode ser usado para conectar dados com outras tabelas.

Usuario: `VARCHAR(32) UNIQUE`. Armazena o nome do usuário em até 32 caracteres e previne que dois usuários tenham o mesmo nome de usuário.

Anexo

A primeira coisa que devemos fazer é “matar” o MySQL:

```
killall mysqld
```

Agora vamos ativar o modo seguro:

```
/usr/bin/mysqld_safe --skip-grant-tables &
```

Vamos entrar nele:

```
/usr/bin/mysql
```

Execute os comandos abaixo:

```
mysql> use mysql;mysql> update user set password = password('SUANOVA SENHA') where user='root'  
and host='localhost';mysql> exitmysqladmin -u root -p shutdown
```

Pronto!

Anexo

Caso ainda não tenha, criar senha de root no MySQL:

```
mysqladmin -u root password SUANOVASENHA
```

Você pode também, renomear o usuário root:

```
mysql> UPDATE user set user='novo-nome-de-usuario' where  
user='root';mysql> exit
```